SENTIMENT ANALYSIS ON DRUG REVIEWS

DATA 207

Lynda Solis Chavez, Riya Shrestha, Felicia Liu

TABLE OF CONTENTS

01

PURPOSE

What is the purpose of this project, the dataset used

PURPOSE

- Pharmaceutical companies are always in constant competition to create **the best solution possible for different illnesses and disease**
- **Problem Statement:**
  - How can pharmaceutical companies best find out how patients feel about their drugs?
- Our goal was to run *sentiment analysis* on drug reviews to **predict if the general sentiment of the drug is positive or negative**
- Knowing how patients feel about certain drugs can **help pharmaceutical companies improve their drugs** if they wish to do so
  - Additional clinical trials to reduce negative side effects

Almost half of all Americans used at least one prescription drug in the past 30 days

according to the CDC

Global pharmaceutical research and development spending is expected to top $200 billion in 2023

according to IFPMA

There are over 19,000 prescription drug products approved for marketing by the FDA

according to the FDA

DRUG REVIEW DATASET

- UCI Machine Learning Repository
- Attributes:
  - drugName
  - Condition
  - Review
  - rating (scale 1-10)
  - Date
  - usefulCount
- 215,063 instances

Felix Gräßer, Surya Kallumadi, Hagen Malberg, and Sebastian Zaunseder. 2018. Aspect-Based Sentiment Analysis of Drug Reviews Applying Cross-Domain and Cross-Data Learning. In Proceedings of the 2018 International Conference on Digital Health (DH '18). ACM, New York, NY, USA, 121-125.

WHY SENTIMENT ANALYSIS?

FEELINGS

Subjective opinions drive many choices

TRENDS

Positive and negative trends can be captured to gain insight to customers

VALUE

Quantifying emotional value is key to making a successful product

02

MODELS

The models used and the problems encountered

MODELS

Linear SVC

Binary Decision Tree

Logistic Regression

Embeddings

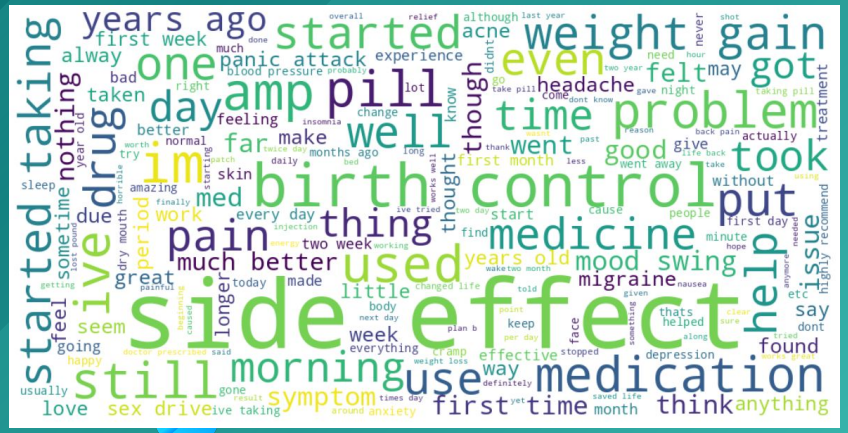
MODEL INPUTS

- X input: Reviews
  - Pre-processed to remove stop words, remove punctuation, remove leading/trailing whitespaces, made lowercase, removed NAs, and stemmed
  - Next the data was processed 2 ways for different models
    - For the **classifier models,** the data was run through a TF-IDF vectorizer
    - For the **embeddings models**, the data was tokenized and padded

- Y input: Binary (0 for negative review or 1 for positive)
  - Reviews rated 1-4 were given a 0
  - Reviews 5 & 6 were deemed neutral and removed from the dataset (original label was -1)
  - Reviews 7-10 were given a 1

HOW WAS THE DATA SPLIT

TRAIN	VALIDATION	TEST
75% of total data	18.75% of total data	6.25% of total data
161,297	40,325	13,441



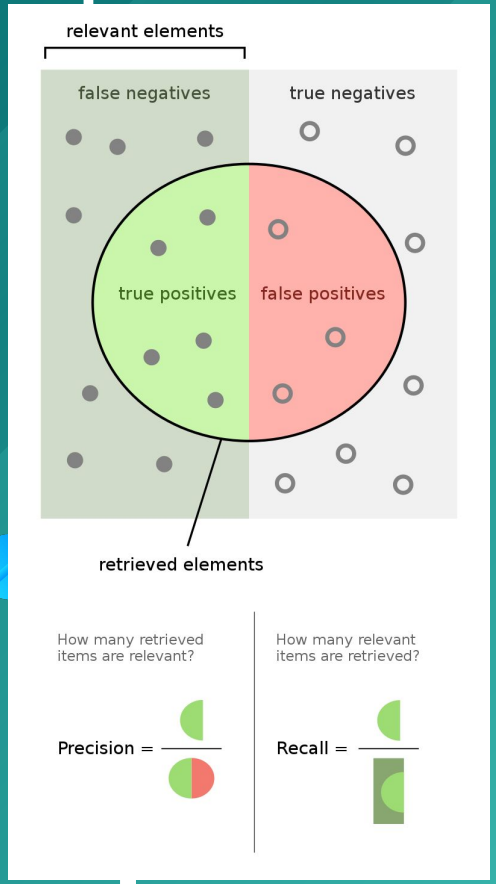
What is TF-IDF?

Term Frequency-Inverse Document Frequency

- Measures how relevant a word is in a document

$$idf\,(t, D) = \log\left(\frac{N}{count\,(d \in D : t \in d)}\right)$$

Image Source:
https://monkeylearn.com/blog/what-is-tf-idf/



What is an F-score?

$$F_1 = \frac{2}{\mathrm{recall}^{-1} + \mathrm{precision}^{-1}} = 2\frac{\mathrm{precision} \cdot \mathrm{recall}}{\mathrm{precision} + \mathrm{recall}} = \frac{2\mathrm{tp}}{2\mathrm{tp} + \mathrm{fp} + \mathrm{fn}}.$$

- Used in the statistical analysis of binary classification
- F1 is the harmonic mean of the precision and recall
- Good for when there are uneven classes sizes like shown in our histogram below

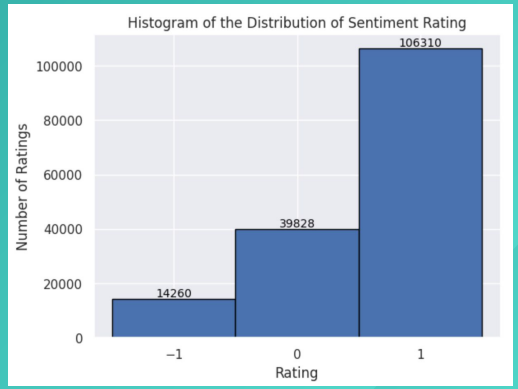


Image Source: https://en.wikipedia.org/wiki/F-score

TF-IDF: Linear SVC

Linear Support Vector Classification

- Measures how relevant a word is in a document

MODEL 1

Linear SVC

ACCURACY

89.3%


```
{'0': {'precision': 0.8330065359477125,
  'recall': 0.7629452259802454,
  'f1-score': 0.7964380565536634,
  'support': 10023},
 '1': {'precision': 0.913056206088993,
  'recall': 0.9421181801019445,
  'f1-score': 0.9273595599576311,
  'support': 26485},
 'accuracy': 0.8929275775172565,
 'macro avg': {'precision': 0.8730313710183527,
  'recall': 0.8525317030410949,
  'f1-score': 0.8618988082556472,
  'support': 36508},
 'weighted avg': {'precision': 0.8910791642399173,
  'recall': 0.8929275775172565,
  'f1-score': 0.8914160344668354,
  'support': 36508}}
```


Image Source: https://www.tutorialspoint.com/scikit_learn/scikit_learn_support_vector_machines.htm

TF-IDF: Logistic Regression

MODEL 2

Logistic Regression

ACCURACY

87.6%

- Logistic regression - Logit model (supervised ML technique)
  - Using binary classification for our case
  - Estimates the probability of an event occurring
    - A logit transformation is applied on the odds (probability of failure/success)
    - Logistic function

$$\text{Logit}(pi) = 1/(1+ \exp(-pi))$$

$$\ln(pi/(1-pi)) = \text{Beta\_0} + \text{Beta\_1}*X\_1 + \ldots + B\_k*K\_k$$

    - For binary classifiers
      - P(<0.5) = 0
      - P(>0.5) = 1

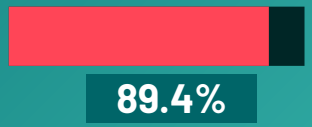

```
{'0': {'precision': 0.8252599243856332,
  'recall': 0.6968971365858525,
  'f1-score': 0.755666143776708,
  'support': 10023},
 '1': {'precision': 0.8916702324917986,
  'recall': 0.9441570700396451,
  'f1-score': 0.9171633442755232,
  'support': 26485},
 'accuracy': 0.8762736934370549,
 'macro avg': {'precision': 0.8584650784387159,
  'recall': 0.8205271033127488,
  'f1-score': 0.8364147440261156,
  'support': 36508},
 'weighted avg': {'precision': 0.8734377760946227,
  'recall': 0.8762736934370549,
  'f1-score': 0.8728254884466741,
  'support': 36508}}
```

TF-IDF: Binary Decision Tree

MODEL 3

Binary Decision Tree

ACCURACY



89.4%

- Supervised machine-learning technique
- Structure based on a sequential decision process
- Subject attributes to a series of binary (yes/no) decisions
- TF-IDF scores are used as a feature
- Conditions of decision tree are TF-IDF weights
- F1 for 0: 80.6%
- F1 for 1: 92.7%


```
{'0': {'precision': 0.8076190476190476,
  'recall': 0.8037513718447571,
  'f1-score': 0.8056805680568058,
  'support': 10023},
 '1': {'precision': 0.9258659028379753,
  'recall': 0.9275438927694921,
  'f1-score': 0.9267041382172092,
  'support': 26485},
 'accuracy': 0.8935575764216063,
 'macro avg': {'precision': 0.8667424752285114,
  'recall': 0.8656476323071246,
  'f1-score': 0.8661923531370075,
  'support': 36508},
 'weighted avg': {'precision': 0.8934021077832116,
  'recall': 0.8935575764216063,
  'f1-score': 0.8934780167173263,
  'support': 36508}}
```

TF-IDF vs. EMBEDDINGS

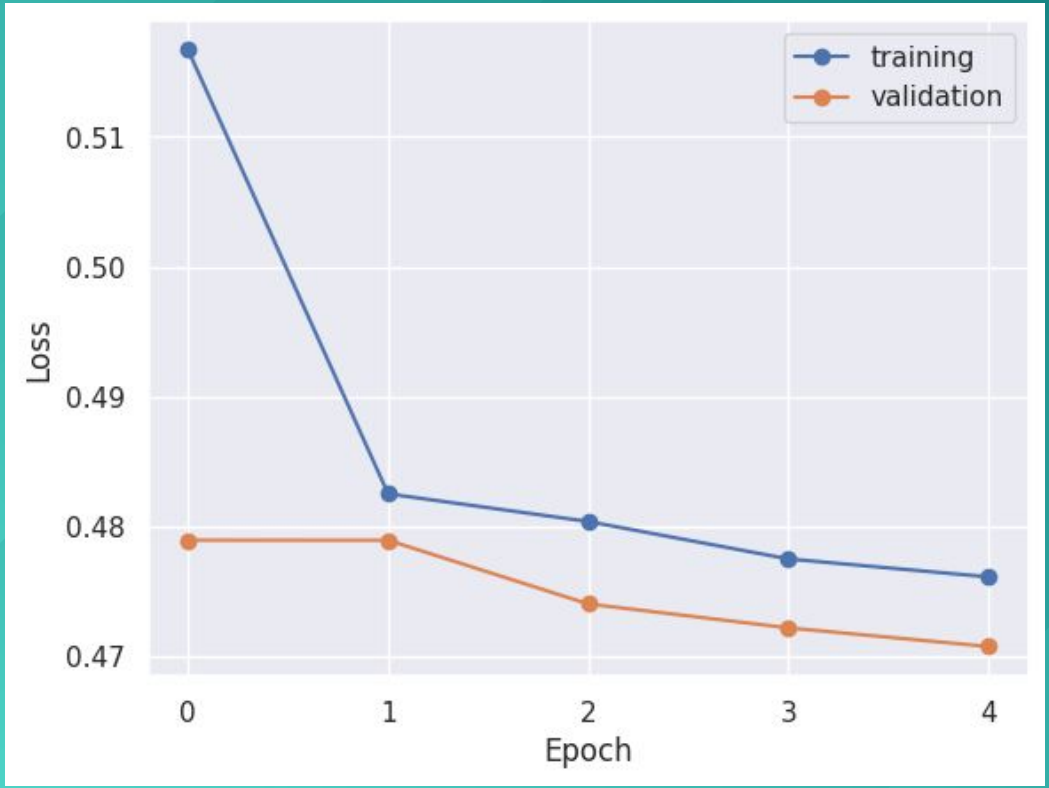
Word Embedding	TF-IDF matrix
Multi dimensional vector which attempts to capture a words relationship to other words	Sparse matrix where each word maps to just a single value, captures no meaning
Often trained on large external corpus	Trained without external data
Must be applied to each word individually	Can be applied to each training document at once
More memory intensive	Less memory intensive
Ideal for problems involving a single word such as a word translation	Ideal for problems with many words and larger document files

Image Source: https://medium.com/@dcameronsteinke/tf-idf-vs-word-embedding-a-comparison-and-code-tutorial-5ba341379ab0

Embedding Hyperparameters

	Training	Validation	Optimizer	Dropout Layer	Extra Dense Layer	Nodes per Dense Layer
Model 1	77.07%	77.16%	Adam	None	None	None
Model 2	72.72%	72.75%	SGD	None	None	None
Model 3	77.50%	77.89%	Adam	Rate = 0.5	Relu	200
Model 4	77.35%	77.58%	Adam	None	Relu	8
Model 5	77.28%	77.47%	Adam	None	Relu	16
Model 6	**77.71%**	**77.89%**	**Adam**	**None**	**Relu**	**32**
Model 7	77.34%	77.42%	Adam	None	Softmax	8
Model 8	77.33%	77.41%	Adam	None	Softmax	16
Model 9	77.31%	77.41%	Adam	None	Softmax	32

Model 3

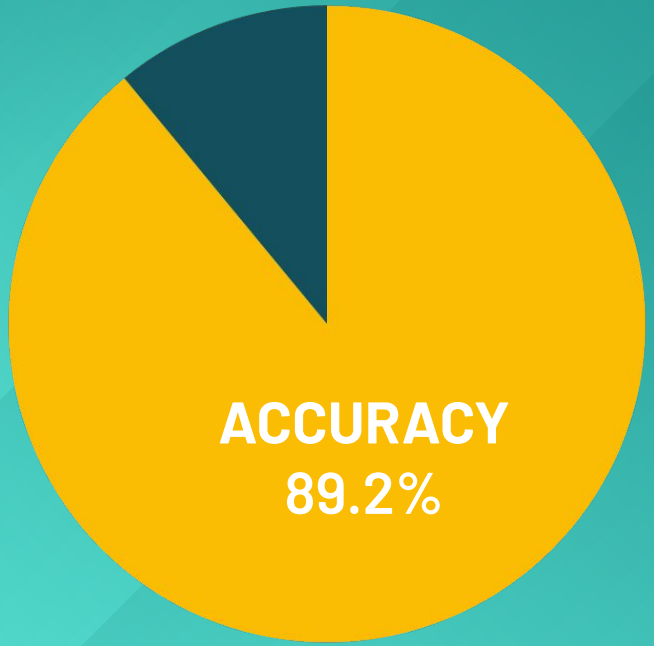


03

CONCLUSION

Conclusion and future steps

Final Model: Binary Decision Tree



F1 score for 0 class = 80.4%
F1 score for 1 class = 92.6%


```
{'0': {'precision': 0.8163141993957704,
  'recall': 0.7935389133627019,
  'f1-score': 0.8047654504839911,
  'support': 3405},
 '1': {'precision': 0.9206546275395033,
  'recall': 0.9306332002281803,
  'f1-score': 0.9256170212765957,
  'support': 8765},
 'accuracy': 0.8922760887428102,
 'macro avg': {'precision': 0.8684844134676368,
  'recall': 0.8620860567954411,
  'f1-score': 0.8651912358802933,
  'support': 12170},
 'weighted avg': {'precision': 0.8914615989586151,
  'recall': 0.8922760887428102,
  'f1-score': 0.8918044001961669,
  'support': 12170}}
```

Future Work

- Oversample and Undersample the imbalanced data
- Neural Networks using TF-IDF
  - Kept crashing when trying to convert matrix to array
- Change Neural Network Hyperparameters

Thank You!

Any Questions?